Crypto Advisor

Agent to the Moon

Brian Hampson

Objectives

- Continuously monitor OHLCV for selected cryptocurrency

- Evaluate technical indicators, did something interesting happen?

- If something interesting happened, deliver relevant analysis and price predictions to inbox

AWS Serverless Architecture

AWS Lambda Dependencies

- Many common libraries NOT included (NumPy, Pandas, CCXT, Keras, TensorFlow)

- What does this mean?

  - You are responsible for building and packaging all dependencies ⚠️

  - Share with Lambda function via **Layer**

- Options

  - **Source distribution:** source code + metadata, compiled on user's machine

  - **Wheels:** pre-compiled binaries, download .whl file directly from PyPI

  - **Pre-built AWS Layers:** reference AWS ARN

  - **Docker Container**

AWS Lambda Dependencies cont.

```
# Layer: numpy_pandas_ccxt_layer
import ccxt
import numpy as np
import pandas as pd
```

ccxt	3/14/2021 1:55 PM	File folder
ccxt-1.43.22.dist-info	3/14/2021 1:55 PM	File folder
cryptography	3/14/2021 1:58 PM	File folder
cryptography-3.4.6.dist-info	3/14/2021 1:58 PM	File folder
numpy	3/14/2021 1:51 PM	File folder
numpy.libs	3/14/2021 1:51 PM	File folder
numpy-1.20.1.dist-info	3/14/2021 1:51 PM	File folder
pandas	3/14/2021 1:42 PM	File folder
pandas-1.2.3.dist-info	3/14/2021 1:41 PM	File folder
pytz	3/14/2021 1:41 PM	File folder
pytz-2021.1.dist-info	3/14/2021 1:41 PM	File folder
requests	3/14/2021 2:01 PM	File folder
requests-2.25.1.dist-info	3/14/2021 2:01 PM	File folder

https://pypi.org/
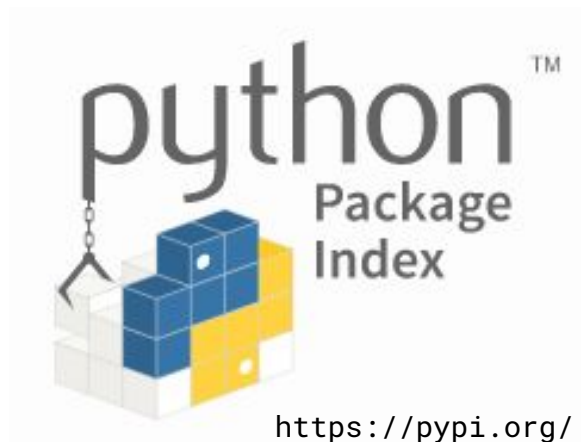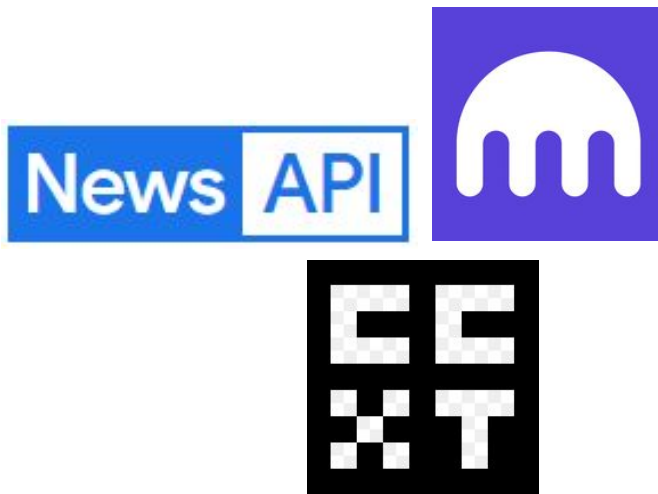
python.zip

Data

What?

- Open High Low Close Volume (OHLCV)
- News articles

Where?

- Kraken API
- News API

How?

- SDKs: CCXT and NewsApiClient
- Data caching on AWS S3

Process and Workflow

1. Develop and test on Google colab

   ○ Better development/test environment

   ○ Identify target libraries

2. Test library dependencies on AWS Lambda

3. Build and deploy packages to Lambda Layers

4. Integrate code from Google colab

   ○ Migrate persistent storage (Google drive to AWS S3)
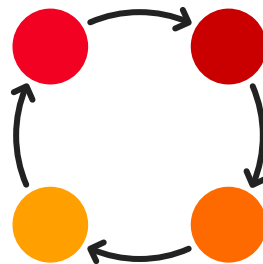
# Notification Triggers

- Technical indicators
  - Exponentially Weighted Moving Average (EWMA)
  - Bollinger Bands
  - Relative Strength Index (RSI)
  - Fibonacci Retracements
  - Moving Average Convergence and Divergence (MACD)
  - Average Directional Index
 - On-Balance Volume (OBV)

Machine Learning

- Price prediction ML models
  - LSTM RNN
  - Linear Regression
  - GARCH
  - Random Forest
- Natural Language Processing (NLP)
  - VADER sentiment analysis
  - Named Entity Recognition
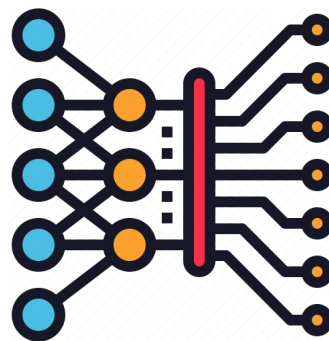
Next steps

- AWS Lambda: Integrate machine learning models

- Add persistent storage with DynamoDB

- Build REST API to encapsulate biz logic

- Build Web Dashboard to provide access on any platform

- Additional technical indicators

- Additional models
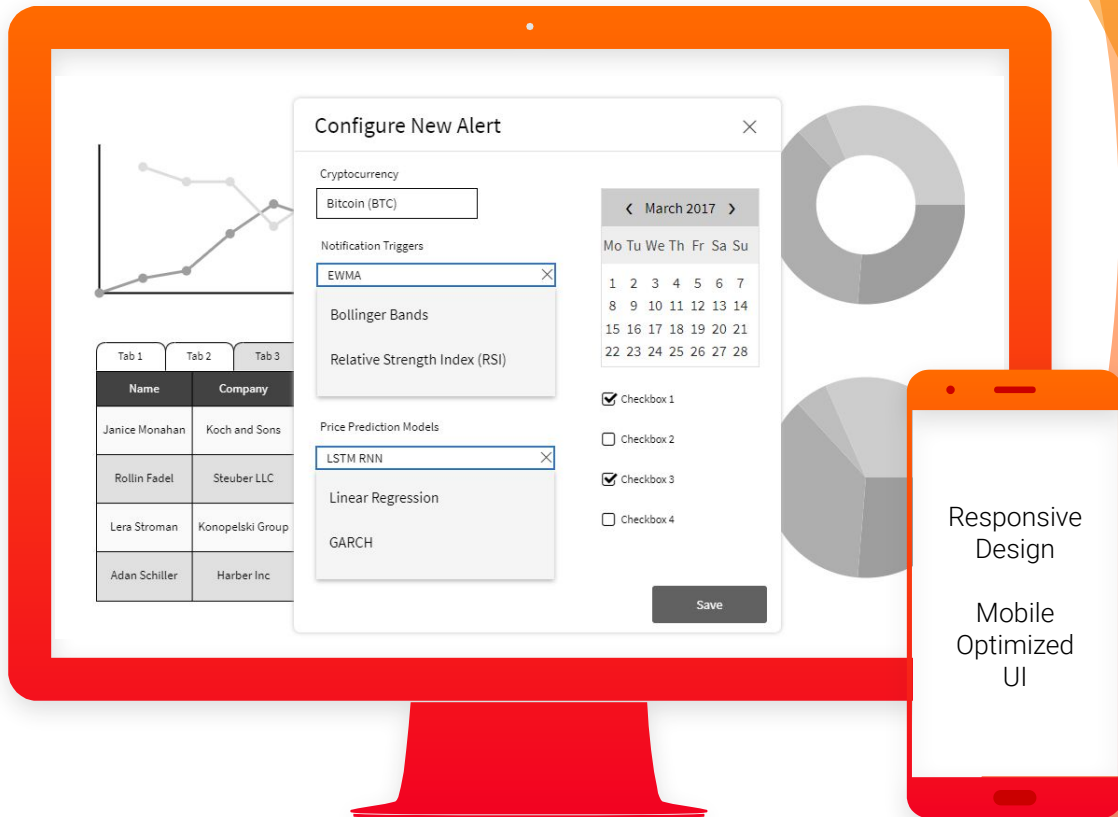
Summary

- AWS serverless architecture for the WIN!

- Achieved end-to-end solution that is both extensible and scalable (and secure)

- Platform for future development.

- Pain points / lessons learned:

    - Difficulties importing Keras and TensorFlow.

    - Use Docker containers for build dependencies.

Questions?